CLAIMS

What is claimed is:

1. A method for managing a code sequence, comprising:

determining first intermediate correlation values for a first plurality of sample sequences during a first clock cycle;

determining second intermediate correlation values for the first plurality of sample sequences during a second clock cycle; and

determining correlation outputs for the first plurality of sample sequences from the first and second intermediate correlation values.

2. The method of Claim 1, wherein determining the first intermediate correlation values comprises processing coefficients in a first code sequence group in parallel with corresponding sample values in corresponding sample sequence groups from the first plurality of sample sequences.

3. The method of Claim 1, wherein determining the second intermediate correlation values comprises processing coefficients in a second code sequence group in parallel with corresponding sample values in corresponding sample sequence groups from the first plurality of sample sequences.

4. The method of Claim 1, wherein determining correlation outputs for the first plurality of sample sequences comprises taking a sum of the first and second intermediate correlation values for each of the first plurality of sample sequences.

5. The method of Claim 1, further comprising:

determining first intermediate correlation values for a second plurality of sample values during a third clock;

determining second intermediate correlation values for the second plurality of sample values during a fourth clock; and

determining correlation output values for the second plurality of sample value from the first and second intermediate correlation values.

6. A method for managing a code sequence, comprising:

processing a first group of coefficients in the code sequence with a first group of sample values in a received sample to determine a first intermediate correlation value during a first clock cycle;

processing a second group of coefficients in the code sequence with a second group of sample values in the received sample to determine a second intermediate correlation value during a second clock cycle; and

determining a correlation output from the first and second intermediate correlation values.

7. The method of claim 6, wherein the code sequence comprises L coefficient and the first and second group of coefficients in the code sequence each comprises n coefficients.

8. The method of claim 7, wherein the first and second group of sample values in the received sample each comprises n sample values.

9. The method of claim 6, wherein the first and second group of coefficients in the code sequence are contiguous.

10. The method of claim 6, wherein the first and second group of sample values in the received sample are contiguous.

11. The method of claim 6, wherein processing the first group of coefficient in the code sequence with the first group of sample values in the received sample comprises determining a sum of the products of the first group of coefficients in the code sequence with the first group of sample values in the received sample.

12. The method of claim 6, wherein processing the second group of coefficients in the code sequence with the second group of sample values in the received sample comprises determining a sum of the products of the second group of coefficients in the code sequence with the second group of sample values in the received sample.

13. The method of claim 6, wherein determining the correlation output from the first and second intermediate correlation values comprises taking the sum of the first and second intermediate correlation values.

14. A method for managing a code sequence, comprising:

organizing the code sequence, having a plurality of contiguous coefficients, into a plurality of contiguous code sequence groups;

selecting a number of sample sequences to process in parallel where each of the sample sequences has contiguous sample values from a received sample;

organizing contiguous sample values from each of a first set of sample sequences to process in parallel into a first set of contiguous sample sequence groups; and

processing coefficients in each of the code sequence groups in parallel with corresponding sample values in corresponding sample sequence groups from the first set of

sample sequences, where each of code sequence groups is processed during a different clock cycle.

15. The method of Claim 14, further comprising:

organizing contiguous sample values from each of a second set of sample sequences to process in parallel into a second set of contiguous sample sequence groups; and

processing coefficients in each of the code sequence groups in parallel with corresponding sample values in corresponding sample sequence groups from the second set of sample sequences, where each of the code sequence groups is processed during a different clock cycle.

16. The method of Claim 14, further comprising:

determining a correlation output for each of the sample sequences; and

determining a synchronization point for the code sequence from the correlation outputs.

17. The method of Claim 16, wherein determining a synchronization output comprises determining a correlation output having a highest numerical value.

18. The method of Claim 14, wherein a first sample value in a first sample sequence includes a first sample value in the received sample and each consecutive sample sequence includes a next contiguous sample value in the received sample as a first sample value the consecutive sample sequence.

19. The method of Claim 14, wherein processing comprises determining a sum of the products of the coefficients in each of the code sequence groups with each of the sample values in corresponding sample sequence groups from the first set of sample sequences.

20. The method of Claim 14, wherein the code sequence comprises a plurality of L contiguous values.

21. The method of Claim 20, wherein the code sequence is organized into a plurality of n code sequence groups.

22. The method of Claim 21, wherein a number, d, sample sequences are selected to process in parallel where each of the sample sequences has L contiguous sample values from the sample.

23. The method of Claim 22, wherein the first set of sample sequences is organized into a plurality of contiguous sample sequence groups having n values each.

24. The method of Claim 14, wherein the code sequence is organized into L/n groups.

25. The method of Claim 14, wherein the processing is completed after L/n clocks.

26. A method for managing a code sequence, comprising:

organizing the code sequence, having L contiguous coefficients, into a number of contiguous code sequence groups having n coefficients each;

selecting a number of sample sequences, d, to process in parallel where each of the sample sequences has L contiguous sample values from a received sample, where a first sample value in a first sample sequence is a first sample value in the received sample and each consecutive sample sequence includes a next contiguous sample value in the received sample as a first sample value in the consecutive sample sequence;

organizing sample values from each of a first set of d sample sequences into a first set of sample sequence groups having n values each; and

processing coefficients in each of the code sequence groups in parallel with corresponding sample values in corresponding sample sequence groups from the first set of d sample sequences, where each of the code sequence groups is processed during a different clock cycle.

27. The method of Claim 26, further comprising:

organizing sample values from each of a second set of d sample sequences into a second set of contiguous sample sequence groups having n values each; andprocessing values in each of the code sequence groups in parallel with corresponding sample values in corresponding sample sequence groups from the second set of d sample sequences, where each of the code sequence groups is processed during a different clock cycle.

28. The method of Claim 26, further comprising:

determining a correlation output for each of the sample sequences; and

determining a synchronization point for the code sequence from the correlation outputs.

29. The method of Claim 28, wherein determining a synchronization output comprises determining a correlation output having a highest numerical value.

30. The method of Claim 26, wherein the code sequence is organized into L/n groups.

31. The method of Claim 26, wherein processing comprises determining a sum of the products of the coefficients in each of the code sequence groups with each of the sample values in corresponding sample sequence groups from the first set of d sample sequences.

32. The method of Claim 26, wherein the processing is completed after L/n clocks.

33. A correlator unit, comprising:

a plurality of code sequence registers that store coefficients from a code sequence group, the plurality of code sequence registers storing coefficients from one code sequence group of a plurality of code sequence groups at a time;

a plurality of sample registers that store sample values from a plurality of sample sequences that are processed in parallel; and

a processing unit that processes coefficients in each of the plurality of code sequence groups in the plurality of code sequence registers in parallel with corresponding sample values in corresponding sample sequence groups from a first plurality of sample sequences in the plurality of sample registers, where each of the code sequence groups is processed during a different clock cycle.

34. The correlator unit of Claim 33, further comprising a plurality of accumulation sub-units each accumulation sub-unit receiving results from the processing unit for a designated sample sequence, each accumulation unit generating a correlation value for the designated sample sequence after each of the code sequence groups are processed.

35. The correlator unit of Claim 33, wherein the processing unit processes the coefficients in each of the plurality of the plurality of code sequence groups in the plurality of code sequence registers in parallel with corresponding sample values in corresponding sample sequence groups from a second plurality of sample sequences in the plurality of sample registers, where each of the code sequence groups is processed during a different clock cycle.

36. The correlator unit of Claim 34, further comprising correlation output processor that determines a synchronization point for the code sequence from the correlation outputs.

37. The correlator unit of Claim 36, wherein the correlation output processor determines a synchronization point from a correlation output having a highest numerical value.

38. The correlator unit of Claim 33, wherein the processing unit determines a sum of products of the coefficients in each of the code sequence groups with corresponding sample values in corresponding

39. A correlator unit, comprising:

a plurality of n code sequence registers that store n coefficients from a code sequence group, the plurality of n code sequence registers storing coefficients from one code sequence group of a plurality of code sequence groups at a time;

a plurality of n+d-1 sample registers that store sample values from a plurality of d sample sequences that are processed in parallel; and

a processing unit that processes coefficients in each of the plurality of code sequence groups in the plurality of n code sequence registers in parallel with corresponding sample values in corresponding sample sequence groups from a first plurality of d sample sequences in the plurality of n+d-1 sample registers, where each of the code sequence groups is processed during a different clock cycle.

40. The correlator unit of Claim 39, further comprising an accumulation sub-unit, corresponding to each of the d sample sequences that are processed in parallel, that receives results from the processing unit for a designated sample sequence and that determines a

correlation output for the designated sample sequence after each of the code sequence groups are processed.

41. The correlator unit of Claim 39, wherein the processing unit processes the coefficients in each of the plurality code sequence groups in the plurality of n code sequence registers in parallel with corresponding sample values in corresponding sample sequence groups from a second plurality of d sample sequences in the plurality of n+d-1 sample registers, where each of the code sequence groups is processed during a different clock cycle.

42. The correlator unit of Claim 40, further comprising correlation output processor that determines a synchronization point for the code sequence from the correlation outputs.

43. The correlator unit of Claim 42, wherein the correlation output processor determines a synchronization point from a correlation output having a highest numerical value.

44. The correlator unit of Claim 39, wherein the processing unit determines a sum of products of the coefficients in each of the code sequence groups with each of the sample values in corresponding sample sequence groups from the first set of d correlation sequences.

45. The correlator unit of Claim 39, wherein the processing is completed after L/n clocks.

46. A correlator unit, comprising:

means for storing coefficients from a code sequence group, the means for storing coefficients storing coefficients from one code sequence group of a plurality of code sequence groups at a time;

means for storing sample values from a plurality of sample sequences that are processed in parallel; and

means for processing coefficients in each of the plurality of code sequence groups in the means for storing coefficients in parallel with corresponding sample values in corresponding sample sequence groups from a first plurality of sample sequences in the means for storing sample values, where each of the code sequence groups is processed during a different clock cycle.